REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER ORC 87-2	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE *(and Subtitle)* An Aggregate Model Of Project-Oriented Production*		5. TYPE OF REPORT & PERIOD COVERED Technical
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Steven T. Hackman Robert C. Leachman		8. CONTRACT OR GRANT NUMBER(s) N00014-76-C-0134
9. PERFORMING ORGANIZATION NAME AND ADDRESS Operations Research Center University of California Berkeley, CA 94720		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS NR 337 015
11. CONTROLLING OFFICE NAME AND ADDRESS Office of Naval Research Department of the Navy Arlington, VA 22297		12. REPORT DATE Febuary 1987
		13. NUMBER OF PAGES 39
14. MONITORING AGENCY NAME & ADDRESS*(if different from Controlling Office)*		15. SECURITY CLASS. *(of this report)* Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT *(of this Report)*

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT *(of the abstract entered in Block 20, if different from Report)*

18. SUPPLEMENTARY NOTES

19. KEY WORDS *(Continue on reverse side if necessary and identify by block number)*

Multi-Project Scheduling
Resource Constrained Project Scheduling
Aggregation of Project Networks
Dynamic Production Functions

20. ABSTRACT *(Continue on reverse side if necessary and identify by block number)*

See Report

AN AGGREGATE MODEL OF

PROJECT-ORIENTED PRODUCTION*

by

Steven T. Hackman[1]

Robert C. Leachman[2]

ORC 87-2   February 1987

An Aggregate Model of Project-Oriented Production

Steven T. Hackman

School of Industrial and Systems Engineering
Georgia Institute of Technology
Atlanta, GA 30332

Robert C. Leachman

Dept. of Industrial Engineering and Operations Research
University of California
Berkeley, CA 94720

ABSTRACT

We consider the problem of dynamically allocating scarce resources to multiple projects in a project-oriented production system such as a naval shipyard. We formulate axioms governing the relationship between resource allocations and work progress of aggregate project activities, from which a model of project execution is derived. This model is more accurate than models which represent aggregate activities in terms of critical path networks or standard linear programming formulations. The model of project execution is then embedded in a resource allocation model suitable for solution by linear programming calculations.

February 26, 1987

1. Introduction

In a *project-oriented production system* a number of large concurrent projects must be carried out subject to inflexible capacities for resources such as skilled labor and equipment. For example, in a naval shipyard as many as 10 ships may be in overhaul at the same time, requiring the careful management of thousands of workers belonging to dozens of skill types. In such organizations, project managers are responsible for keeping projects on schedule and within budget. To avoid project delays and budget overruns, schedules must be developed reflecting efficient and feasible allocations of resources.

We propose a hierarchical approach to scheduling in a project-oriented production system. The first step is to establish overall project timeframes and to allocate resources to project managers. The allocations then serve as capacities for resource-constrained scheduling of individual projects. The advantages of a hierarchical approach are twofold. First, this approach avoids the computational burdens of the traditional approach involving simultaneous resource-constrained scheduling of multiple projects. (See, for example, Kurtulus and Davis [1985].) Second, a hierarchical approach is consistent with the organizational structure: higher levels of management plan overall project timeframes and the allocation of scarce resources among projects, while lower levels are responsible for detailed scheduling of individual projects.

Like detailed project scheduling, the efficient allocation of resources among projects requires a model of project execution, that is, a model expressing project progress as a function of resource allocation. Aggregating detailed activities which utilize similar mixes of resources helps to reduce the size of the allocation problem. Previous research concerning the aggregation of project networks emphasizes serial aggregation to maintain strict precedence. (See, for example, Parikh and Jewell [1965], Eardley [1960], Vlach [1968], Burman

[1972], Archibald [1972], and Harris [1978].) However, in project networks describing project-oriented production, detailed activities in series seldom utilize the same mix of resources, yet parallel activities frequently do. Hence, parallel activities need to be aggregated. Consider, for example, Figure 1. The application of resources to the re-install aggregate is constrained by the application of resources to the repair aggregate. Since the operations of the repair and re-install aggregates can overlap in time, it is inaccurate to model this constraint at the aggregate level as strict precedence. An entirely new model of project execution is needed.

In this paper we formulate an aggregate-level model of execution of an industrial project. The problem of modeling the appropriate constraint on the resource applications to serial aggregates is characterized as the problem of mapping resource applications at one aggregate into outputs used by successor aggregates, and then expressing the appropriate inventory balance constraint. In terms of the example, a production function must be developed to describe the supply of repaired equipment as a function of resource applications at the repair aggregate. Resource applications at the re-install aggregate are then constrained by the supply of repaired equipment.

For the basic structure shown in Figure 1, we prove that the production function is uniquely determined when two reasonable axioms are adopted. More complex structures are decomposed as *replications* or *aggregations* of the basic structure. Production functions for complex structures are derived as weighted combinations of instances of the production function modeling the basic structure.

The basic aggregate modeling approach explored in this paper originated with the work of Boysen [1982] and Leachman and Boysen [1982, 1985], who were the first to explicitly model workflow and resource constraints on aggre-

gates of parallel activities. Their model of workflow was only intuitively and empirically justified.

The main contribution of this paper is a formal methodology for developing aggregate constraints describing project execution. In this approach, a continuous-time model of project execution is derived from elementary principles and basic assumptions (axioms). This model is then reduced to a discrete-time computational form. There are two advantages of the approach: First, accurate constraints describing the production process are obtained; and second, by derivation from axioms, we make clear on what basis the production functions modeling aggregate network structures are valid.

2. Modeling Resource Application at Detailed and Aggregate Levels of Detail

We begin by modeling resource applications at detailed and aggregate activities following Leachman and Boysen [1985]. Each project in the production system is modeled by a standard resource-constrained activity-on-node critical path network. The precedence network is an acyclic directed graph on L nodes. The set of arcs is denoted by the symbol H. An arc from node (activity) l to node (activity) m indicates that activity m cannot start until activity l has finished. As notation let ES_l denote the early-start time for activity l, LS_l denote the late-start time for activity l, d_l denote the duration of activity l, and let r_l^k denote the total amount of resource k that activity l requires, $k = 1, 2, \ldots, K$. We assume that the LS_l's are based on a resource-feasible finish time for the project.

At the detailed level, resources are assumed to be applied to an activity at constant rates between start and finish of the activity. That is, if activity l starts at time S_l, $ES_l \leq S_l \leq LS_l$, then between time S_l and time $S_l + d_l$ activity l loads each resource k at the rate $\dfrac{r_l^k}{d_l}$. Let $y_l^k(\tau)$ denote the application of

resource k at time τ by activity l. The critical path assumptions imply that

$$y_l^k(\tau) = r_l^k z_l(\tau), \quad k = 1, 2, \ldots, K, \tag{2.1}$$

where

$$z_l(\tau) = \begin{cases} \dfrac{1}{d_l} & \text{if } \tau \in (S_l, S_l + d_l] \\ 0 & \text{otherwise.} \end{cases} \tag{2.2}$$

The function $z_l(\tau)$ is called the *operating intensity* for activity l. The cumulative intensity $Z_l(t) = \int_0^t z_l(\tau) d\tau$ measures the fraction of the resources required by activity l which has been applied up to time t.

The aggregate level is also modeled by a network of activities labeled $A_1, \ldots, A_N$. Each A_i is an aggregation of a number of parallel, detailed critical path activities. Each detailed activity l is assigned to exactly one aggregate A_i, indicated by $l \in A_i$. An arc between aggregates A_i and A_j exists in the aggregate network if there exists an arc $(l, m) \in H$ such that $l \in A_i$ and $m \in A_j$. Hereafter, we shall use the same symbol to denote corresponding functions at detailed and aggregate levels, using the subscripts "l" or "m" for a detailed-level function and using subscripts "i" or "j" for an aggregate-level function. Functions denoted with capital letters are cumulative functions.

Aggregate activities are formed only if the detailed activities within the aggregate utilize the same *mix* of resources: that is, the ratios $\dfrac{r_l^k}{\sum\limits_{l \in A_i} r_l^k}$ are independent of k. For industrial project networks the resource-mix requirement on aggregation is not restrictive. In such networks, there are many parallel activities using very similar or identical mixes of resources.

We illustrate the aggregation with the simple example shown in Figure 1. Relevant numerical data is provided in Table 1. The four activities which comprise the repair aggregate each utilize the same mix of resources; likewise,

the four activities comprising the re-install aggregate have in common another mix of resource requirements. In an industrial network, there would be tens or scores of parallel repair activities utilizing the same mix, and tens or scores of re-install activities all utilizing another mix.

Let α_{li} denote activity's l's percentage of each resource consumed by A_i, let r_i^k denote the total amount of resource k that aggregate A_i requires, and let $y_i^k(\tau)$ denote the application through time of resource k to aggregate A_i. Since $y_i^k(\tau) = \sum_{l \in A_i} y_l^k(\tau)$ and by assumption $y_l^k(\tau) = r_l^k z_l(\tau)$ for some operating intensity z_l, it follows that $y_i^k(\tau) = \sum_{l \in A_i} r_l^k z_l(\tau)$. The resource-mix requirement further implies that

$$y_i^k(\tau) = \left(\sum_{l \in A_i} r_l^k \right) \left[\sum_{l \in A_i} \frac{r_l^k}{\left(\sum_{l \in A_i} r_l^k \right)} z_l(\tau) \right] = r_i^k \left[\sum_{l \in A_i} \alpha_{li} z_l(\tau) \right] = r_i^k z_i(\tau) \qquad (2.3)$$

where $z_i(\tau) = \sum_{l \in A_i} \alpha_{li} z_l(\tau)$. Thus when the resource-mix requirement is enforced an aggregate activity's resource applications may be indexed by one profile $z_i(\tau)$ which is a *convex* combination of the intensities of the detailed activities. (Compare (2.3) with (2.1).) Each *aggregate operating intensity* is *induced* from a schedule for the detailed activities within aggregate A_i. Similar to the detailed intensity function $z_l(\tau)$, the aggregate intensity function $z_i(\tau)$ measures progress towards completion in so far as the *cumulative* intensity up to time t, $Z_i(t) = \int_0^t z_i(\tau) d\tau$, represents the fraction of the total resources applied by time t required to complete all of the detailed activities within aggregate A_i.

Let z_i^L denote the operating intensity for aggregate A_i induced by the late-start schedule, and let z_i^E denote the operating intensity for aggregate A_i induced by the early-start schedule. By definition of early- and late-start schedules, all induced Z_i's satisfy

$$Z_i^L(t) \le Z_i(t) \le Z_i^F(t) \quad \text{for all } t. \tag{2.4}$$

The "boundary curves" Z_i^L and Z_i^F provide lower and upper *bounds through time* on the cumulative use of resources by aggregate A_i. The boundary curves for the repair and re-install aggregates are graphed in Figure 2. Let $E_i = \min_{l \in A_i} ES_l$ and let $L_i = \max_{l \in A_i} LS_l + d_l$. The interval of time over which an aggregate can be operating (i.e. applying resources) is given by $[E_i, L_i]$. In the example $[E_i, L_i] = [0,10]$. For each aggregate A_i it is assumed that $Z_i^F(t) > Z_i^L(t)$ for all $t \in (E_i, L_i)$.

3. A Structured, Formal Development of an Aggregate Model

Even before capacities of resources are considered, the feasible choices for intensities of the aggregate activities are limited. In the example it is obvious that the feasible choices for the applications of resources by the repair and re-install aggregates are *dependent*, i.e., the choice Z_{A_i} for operating the repair aggregate restricts the choice Z_{A_j} for operating the re-install aggregate. The strict precedence model of critical path methods insists that applications by predecessors must be complete before applications to any follow-on activities may commence. Since the periods of operation of the repair and re-install aggregates typically overlap, a different model of workflow must be developed.

3.1. Formulation of the Aggregate Activity Dependence Relationships

Our structured approach characterizes the problem of developing a dependence relationship between A_i and A_j as a problem of developing a *dynamic production function* mapping resource applications at A_i into outputs from A_i, and then expressing conservation through time of the product produced by A_i and input to A_j. To make our approach clear, we first reformulate resource-constrained CPM as a model of production, whereby strict precedence between

two detailed activities is expressed as a form of inventory balance, as follows. For each activity l let σ_l denote the set of all possible intensity functions, i.e., those defined by (2.2) with $ES_l \leq S_l \leq LS_l$, and let Σ_l denote the corresponding set of all possible cumulative intensities. For each $(l,m) \in H$ define $f_l^{(l,m)}: \sigma_l \to \sigma_m$ as follows:

$$f_l^{(l,m)}(z_l)(\tau) = \begin{cases} \dfrac{1}{d_m} & \text{if } \tau \in (S_l + d_l, S_l + d_l + d_m] \\ 0 & \text{otherwise.} \end{cases}$$

(Clarifying the notation, z_l and $f_l^{(l,m)}(z_l)$ are both functions of time. $f_l^{(l,m)}(z_l)(\tau)$ denotes the value of the latter function at time τ.) Referring to Figure 3a, the functional $f_l^{(l,m)}$ maps the intensity curve for activity l into the *earliest* intensity for activity m consistent with the assumption of strict precedence. Let $F_l^{(l,m)}(t) = \int_0^t f_l^{(l,m)}(z_l)(\tau) d\tau$ denote the cumulative function corresponding to $f_l^{(l,m)}(z_l)$. The constraint

$$F_l^{(l,m)}(Z_l)(t) \geq Z_m(t) \quad \text{for all } t \tag{3.1}$$

is mathematically equivalent to the requirement that $S_l + d_l \leq S_m$. See Figure 3b. Thus (3.1) expresses strict precedence via a *functional relationship* between the resource applications at l and m indexed by z_l and z_m, respectively. From the perspective of dynamic production theory (Hackman and Leachman (1986)), (3.1) has a natural interpretation as an inventory balance constraint: $f_l^{(l,m)}(z_l)$ is a (dynamic) production function representing the "output" of product "(l,m)" by activity l, and z_m represents the application of intermediate product (l,m) by activity m.

Conceptually, our model of resource use at the aggregate level will parallel our model of resource use at the detailed level. We view aggregate activities as producing intermediate products used by follow-on aggregate activities. Pro-

duction by an aggregate activity is explicitly modeled via a (dynamic) *aggregate production function* $F_i^{(i,j)}: \Sigma_i \rightarrow \Sigma_j$ similar in spirit to $F_l^{(l,m)}(Z_l)$. The feasible set of choices for resource applications of aggregate activities A_i and A_j is expressed via the inventory balance constraint

$$F_i^{(i,j)}(Z_i)(t) \geq Z_j(t) \quad \text{for all } t \tag{3.2}$$

similar in spirit to (3.1). Throughout the remainder of this section we shall suppress the superscript (i,j) and denote $F_i^{(i,j)}(Z_i)$ by $F_i(Z_i)$.

The ideal choice for $F_i(Z_i)$ is given by

$$\sum_{l \in A_i} \alpha_{mj} F_l^{(l,m)}(Z_l). \tag{3.3}$$

(3.3) is precisely the cumulative aggregate operating intensity for A_j induced by the earliest schedule for the activities within A_j consistent with the finish times for the activities in A_i which precede them. Graphed in Figure 4(b) is the ideal output curve corresponding to the input curve Z_i^* graphed in Figure 4(a). Unfortunately, the model for F_i given in (3.3) is unworkable: it incorporates knowledge of the schedules (start-times) for the activities within A_i. A model for the aggregate production function must be *independent* of such knowledge so as to not defeat the whole point of aggregation. We shall refer to the function defined in (3.3) as the *ideal* aggregate production function since it is the logical choice if one knew the detailed information. The goal now is to develop an appropriate approximation to the ideal aggregate production function defined in (3.3).

The first step is to approximate the domain Σ_i. Since each $Z_i \in \Sigma_i$ satisfies (2.4) we therefore approximate Σ_i by the set of all non-decreasing continuous curves satisfying (2.4). We denote this set by the symbol $\widehat{\Sigma}_i$. Since $\Sigma_i \subset \widehat{\Sigma}_i$ no feasible choice has been eliminated. Our next goal is to define (model) a function $F_i: \widehat{\Sigma}_i \rightarrow \widehat{\Sigma}_j$ which "reasonably" approximates the ideal aggregate production

function defined on Σ_i yet is still tractable for analysis. Our method is to establish axioms stating reasonable properties that F_i should satisfy, and then derive functions which satisfy them.

3.2. An Axiomatic Model of the Production Function

The first property we impose on our choice for F_i is best motivated via our example. Figure 4(a) shows the graph of the aggregate operating intensity Z_i^* for our example. Most of the activities within A_i are starting near their early-start times. Note that when an activity is "running early" the ratio of the shaded area shown in Figure 4(a) to the area between the boundary curves of A_i is large (close to 1). Graphed in Figure 5(a) is an arbitrarily proposed cumulative output curve resulting from the input curve Z_i^*. For this proposed output curve, the inventory balance constraint (3.2) restricts the choices for Z_j to the shaded area shown in Figure 5(a), which is small (close to 0) compared to the area between the boundary curves of A_j. Since the relative area is small the inventory balance constraint is *forcing* A_j to essentially "run late", yet A_i is running early. The cumulative output in Figure 5(a) thus provides an unreasonable bound on the choices for Z_j given the choices for Z_i^*. Furthermore, comparing Figures 4(b) and 5(a) we can see that the proposed output curve is not very "close" to the ideal output curve.

On the other hand, the cumulative output curve plotted in Figure 5(b) is a much better choice for $F_i(Z_i^*)$, as it appears to be running as early in the "window" of A_j as Z_i^* is running in the "window" of A_i. Comparing Figures 4(b) and 5(b), we can see that the output curve shown in Figure 5(b) is a very good approximation to the ideal output curve. Note that the proportion of area between the Z_j^F and Z_j^L curves which is below the $F_i(Z_i^*)$ curve in Figure 5(b) corresponds to the proportion of area between the Z_i^F and Z_i^L curves which is below the Z_i^* curve. The implied property on F_i may be expressed as

$$\frac{\int_{E_j}^{L_j}[F_i(Z_i)(x) - Z_j^L(x)]dx}{\int_{E_j}^{L_j}[Z_j^F(x) - Z_j^L(x)]dx} = \frac{\int_{E_i}^{L_i}[Z_i(x) - Z_i^L(x)]dx}{\int_{E_i}^{L_i}[Z_i^F(x) - Z_i^L(x)]dx}. \tag{3.4}$$

However, (3.4) is not sufficient, since (3.4) provides no restriction on the *distribution* of the area below the curve $F_i(Z_i)$. The reasonable proposal for $F_i(Z_i^*)$ graphed in Figure 5(b) has the property that as time t ranges from E_j to L_j, the distribution of the relative area below $F_i(Z_i)$ *continuously* reflects the distribution of the relative area below Z_i. The extension of (3.4) that we require is expressed as

$$\frac{\int_{E_j}^{t}[F_i(Z_i)(x) - Z_j^L(x)]dx}{\int_{E_j}^{t}[Z_j^F(x) - Z_j^L(x)]dx} = \frac{\int_{E_i}^{\rho_{ji}(t)}[Z_i(x) - Z_i^L(x)]dx}{\int_{E_i}^{\rho_{ji}(t)}[Z_i^F(x) - Z_i^L(x)]dx} \quad \text{for all } t \in [E_j, L_j] \tag{3.5}$$

where $\rho_{ji}: [E_j, L_j] \to [E_i, L_i]$ is some differentiable, increasing, 1-1, onto transformation of time satisfying $\rho_{ji}(E_j) = E_i$ and $\rho_{ji}(L_j) = L_i$. The transformation of time is necessary for (3.5) to make sense since relative area up to a point in time in the interval of operation for A_i must be compared to relative area up to a corresponding point in time in the interval of operation of A_j.

Each F_i which satisfies (3.5) maps the boundary curves of A_i onto the boundary curves of A_j:

$$F_i(Z_i^F) = Z_j^F \tag{3.6i}$$
$$F_i(Z_i^L) = Z_j^L. \tag{3.6ii}$$

In addition, each F_i is *monotone:*

$$\text{if } Z_i^1(t) \geq Z_i^2(t) \text{ for all } t, \text{ then } F_i(Z_i^1)(t) \geq F_i(Z_i^2)(t) \text{ for all } t. \tag{3.7}$$

Both (3.6) and (3.7) are properties of the ideal aggregate production function.

We summarize our first axiom as

Axiom AI (Area Interpolation). The aggregate production function $F_i : \hat{\Sigma}_i \rightarrow \hat{\Sigma}_j$ must satisfy (3.5) where $\rho_{ji} : [E_j, L_j] \rightarrow [E_i, L_i]$ is a differentiable, increasing, 1-1, onto transformation of time satisfying $\rho_{ji}(E_j) = E_i$ and $\rho_{ji}(L_j) = L_i$.

A closed form expression for $F_i(Z_i)$ can be derived from Axiom AI, as follows. To simplify the notation for each aggregate A_i let $H_i(x) = Z_i^E(x) - Z_i^L(x)$ denote the height between the boundary curves at time x; further we shall suppress the subscripts on the time transformation and write $\rho(t)$ in lieu of $\rho_{ji}(t)$. Let $R_\rho(t)$ denote the ratio of the area between the boundary curves of A_j up to time t to the area between the boundary curves of A_i up to time $\rho(t)$, i.e.,

$$R_\rho(t) = \frac{\int_{E_j}^{t} H_j(x)\,dx}{\int_{E_i}^{\rho(t)} H_i(x)\,dx}. \tag{3.8}$$

Upon re-arrangement (3.5) becomes

$$\int_{E_j}^{t} F_i(Z_i)(x)\,dx = \int_{E_j}^{t} Z_j^L(x)\,dx + R_\rho(t)\left[\int_{E_i}^{\rho(t)} \{Z_i(x) - Z_i^L(x)\}\,dx\right]. \tag{3.9}$$

Differentiate each side of (3.9) with respect to t to obtain

$$F_i(Z_i)(t) = Z_j^L(t) + R_\rho(t)\{Z_i(\rho(t)) - Z_i^L(\rho(t))\}\rho'(t) +$$
$$R_\rho'(t)\left[\int_{E_i}^{\rho(t)} \{Z_i(x) - Z_i^L(x)\}\,dx\right] \tag{3.10}$$

where

$$R_\rho'(t) = \frac{\left[\int_{E_i}^{\rho(t)} H_i(x)\,dx\right]H_j(t) + \left[\int_{E_j}^{t} H_j(x)\,dx\right]H_i(\rho(t))\rho'(t)}{\left[\int_{E_i}^{\rho(t)} H_i(x)\,dx\right]^2}. \tag{3.11}$$

For a choice of ρ, F_i is given by (3.10) with $R_\rho'(t)$ given by (3.11).

To specify a particular choice for F_i it remains to specify a reasonable choice for ρ. We derive our choice for ρ by restricting the functional form of F_i, as follows:

$$F_i(Z_i)(t) = h_t(Z_i(\rho(t))) \quad \text{for all } t \in [E_j, L_j] \tag{3.12}$$

for some class of functions $\{h_t\}$, $t \in [E_j, L_j]$. The functional form for F_i expressed in (3.12) restricts the cumulative output curve at time t to be *solely* a function of the cumulative resource applications at time $\rho(t)$, as indexed by $Z_i(\rho(t))$. Note that (3.12) is a generalization of the familiar use of time lags in production planning models, in that both the lag $\rho(t)$ and the input-output transformation h_t can be time-varying. In terms of the example, re-installation is allowed to be $100[h_t(0.7)]\%$ complete at time t if repairs were 70% complete at the earlier time $\rho(t)$.

We summarize our second axiom as

Axiom TL (Time Lag). The aggregate production function $F_i : \hat{\Sigma}_i \to \hat{\Sigma}_j$ is of the form (3.12) for some class of functions h_t, $t \in [E_j, L_j]$, that is consistent with the domain and range of F_i.

Assuming F_i satisfies both axioms (AI and TL), we now show that *both* ρ and h_t are *uniquely* determined, and hence so is F_i. A quick glance at (3.10) shows that the term $\int_{E_i}^{\rho(t)} \{Z_i(x) - Z_i^L(x)\} dx$ is a function of $Z_i(\tau)$ for $\tau < \rho(t)$. From axiom TL, it follows that $R_\rho{}'(t) = 0$, i.e., $R_\rho(t)$ is constant in t. Setting $t = L_j$ in (3.8),

$$R_\rho(t) = \frac{\int_{E_j}^{t} H_j(x) dx}{\int_{E_i}^{\rho(t)} H_i(x) dx} = \frac{\int_{E_j}^{L_j} H_j(x) dx}{\int_{E_i}^{L_i} H_i(x) dx} \equiv R \quad \text{for all } t$$

or equivalently

$$\frac{\int\limits_{E_j}^{t} H_j(x)dx}{\int\limits_{E_j}^{L_j} H_j(x)dx} = \frac{\int\limits_{E_i}^{\rho(t)} H_i(x)dx}{\int\limits_{E_i}^{L_i} H_i(x)dx} \quad \text{for all } t. \tag{3.13}$$

Thus ρ is unique, implicitly defined by (3.13). A vertical slice at time $\rho(t)$ divides the area between the boundary curves of A_i into the same proportion as the vertical slice at time t divides the area between the boundary curves of A_j. (See Figure 6.) Since $R_\rho(t) \equiv R$, we can simplify (3.10) to

$$F_i(Z_i)(t) = h_t(Z_i(\rho(t))) = [(R)\rho'(t)]Z_i(\rho(t)) + \{ Z_j^L(t) - [(R)\rho'(t)]Z_i^L(\rho(t)) \}. \tag{3.14}$$

Thus the h_t's are *necessarily* linear. Differentiating both sides of (3.13), we obtain

$$\rho'(t) = \frac{1}{R} \cdot \frac{H_j(t)}{H_i(\rho(t))}. \tag{3.15}$$

Substituting (3.15) into (3.14),

$$F_i^{(i,j)}(Z_i)(t) = Z_j^L(t) + \frac{H_j(t)}{H_i(\rho_{ji}(t))}[Z_i(\rho_{ji}(t)) - Z_i^L(\rho_{ji}(t))], \quad \text{for all } t, \tag{3.16}$$

and then substituting (3.16) into the inventory balance constraint (3.2) yields

$$\frac{Z_i(\rho_{ji}(t)) - Z_i^L(\rho_{ji}(t))}{H_i(\rho_{ji}(t))} \geq \frac{Z_j(t) - Z_j^L(t)}{H_j(t)} \quad \text{for all } t. \tag{3.17}$$

We summarize our model development with the following Theorem.

Theorem: If $F_i^{(i,j)}: \widehat{\Sigma}_i \to \widehat{\Sigma}_j$ satisfies Axioms AI and TL, then $F_i^{(i,j)}$ is uniquely defined by (3.16) where ρ_{ji} is implicitly defined by (3.13).

3.3. Remarks

The dependence constraint proposed for the basic aggregate network

structure studied in this section is (3.17), which implicitly defines an aggregate production function F_i (3.14) satisfying axioms AI and TL. Graphed in Figure 5(b) is the cumulative output curve corresponding to this production function. As noted earlier, a comparison of Figures 4(b) and 5(b) shows that $F_i(Z_i^*)$ is very close to the ideal output curve.

To develop a computational form for the model expressed in (3.17), we divide time into discrete intervals $(0,1], \ldots, (t-1,t], \ldots, (T-1,T]$ and define intensity variables $z_i(\tau)$ representing a constant intensity rate for A_i during the interval $(\tau, \tau-1]$. We enforce (3.17) at the integer points in time. The constraint (3.17) can be expressed in terms of the discrete-time variables as follows: For a real number x, let x^+ denote the smallest integer greater than or equal to x, and let x^- denote the largest integer less than or equal to x. Each Z_i is a piecewise-linear curve, expressed in terms of the variables as

$$Z_i(t) = \sum_{\tau=1}^{t^-} z_i(\tau) + (t-t^-)\{z_i(t^+)\}.$$

It may be readily verified that

$$Z_i^I(t) = Z_i^I(t^-) + (t-t^-)\{Z_i^I(t^+) - Z_i^I(t^-)\},$$

and similarly,

$$H_i(t) = H_i(t^-) + (t-t^-)\{H_i(t^+) - H_i(t^-)\}.$$

To compute $\rho_{ji}(t)$, let

$$A_i(t) = \frac{\displaystyle\int_{E_i}^{t} H_i(x)dx}{\displaystyle\int_{E_i}^{L_i} H_i(x)dx}$$

denote the relative area between the boundary curves of A_i. By assumption $H_i(x)$ is strictly positive on (E_i, L_i), so that the curve $A_i(t)$ is strictly increasing on (E_i, L_i) and hence has an inverse. The implicit definition (3.13) for ρ_{ji} is

simply the statement that $\rho_{ji}(t) = A_i^{-1}(A_j(t))$. It may be readily verified that

$$A_i(t) = A_i(t^-) + (t-t^-)H_i(t^-) + \frac{1}{2}(t-t^-)\{H_i(t^-) + (t-t^-)[H_i(t^+) - H_i(t^-)]\}. \quad (3.18)$$

If $A_i(t)$ and $A_j(t)$ are precomputed for all integer t, then given an arbitrary t, the corresponding time point $\rho_{ji}(t)$ for which $A_i(\rho_{ji}(t)) = A_j(t)$ can be easily computed using (3.18). By precomputing $Z_i^L(t)$, $Z_j^L(t)$, $H_i(t)$, $H_j(t)$ and $\rho_{ji}(t)$ for all integer t, linear inequalities in the $z_i(\tau)$ variables can be constructed to enforce (3.17).

We note that it is not necessary that $F_i(Z_i)$ satisfy both axioms AI and TL for it to be computationally practical. The Area Interpolation Axiom (3.10) reduces to a linear expression for $F_i(Z_i)(t)$ in terms of the variables $z_i(1)$, $z_i(2)$, ...,$z_i(\rho_{ji}(t)^-)$, $z_i(\rho_{ji}(t)^+)$. (The weights of the linear expression are non-linear functions of t.) If axiom TL is not adopted, then by the theorem we know that some ρ_{ji} other than (3.13) must be specified.

Even if the detailed arcs from A_i to A_j do not provide a one-to-one correspondence between detailed activities in A_i and A_j, the model for $F_i(Z_i)$ expressed in (3.16) still applies. In fact, the model for $F_i(Z_i)$ does not preclude strict precedence among detailed activities within an aggregate. However, the resource-mix requirement typically would not hold for such aggregation.

Finally, we remark that both axioms are testable. For various intensities $Z_i \in \Sigma_i$ one could measure how $F_i(Z_i)$ deviates from the ideal output curve. In limited simulation experiments, Dalebout [1983] shows that the model performs reasonably well. In general, simulation tests conducted *a priori* could provide a means of evaluating whether or not an acceptable aggregation of a project network structure has been performed.

4. More Complex Aggregate Subnetworks

In this section we develop the inventory balance constraints of the aggregate model for more complex network structures. We demonstrate that various aggregate network structures may be viewed as *replications* or *aggregations* of the basic aggregate network structure.

4.1. Replicated Network Structures

Consider the subnetwork shown in Figure 7(a). The task here is to determine a set of constraints which serve to constrain the possible choices for Z_3 given choices for Z_1 and Z_2. To develop the set of constraints we construct an *equivalent* network structure, and analyze it instead. Consider Figure 7(b). It is obtained by replicating aggregate 3 so that we have aggregates A_3 and $A_{3'}$. The precedence constraints for the network system in 7(b) are *identical* to the original precedence constraints. Note that Figure 7(b) consists of two examples of the basic aggregate network structure discussed in Section 3. Hence, inventory balance constraints for the aggregate subnetwork shown in Figure 7(b) are

$$F_1^{(1,3)}(Z_1) \geq Z_3 \tag{4.1i}$$

$$F_2^{(2,3')}(Z_2) \geq Z_{3'} \tag{4.1ii}$$

$$Z_3 = Z_{3'} \tag{4.1iii}$$

We add (4.1iii) to ensure that aggregates A_3 and $A_{3'}$ are identically operated.

Substituting (4.1iii) into (4.1ii), we see that the inventory balance constraints governing the structure in Figure 7(a) must be

$$F_1^{(1,3)}(Z_1) \geq Z_3 \tag{4.2i}$$

$$F_2^{(2,3)}(Z_2) \geq Z_3 \tag{4.2ii}$$

From (4.2) we see that aggregate A_3 requires two distinct intermediate product inputs, one from aggregate A_1 and the other from aggregate A_3. We remark that the derivation of the abstract system (4.2) is independent of the choice for

the basic aggregate production function. Substituting into (4.2) our choice (3.17) for the basic aggregate production function, constraints (4.2) become

$$\frac{Z_3(t) - Z_3^I(t)}{H_3(t)} \leq \frac{Z_1(\rho_{31}(t)) - Z_1^I(\rho_{31}(t))}{H_1(\rho_{31}(t))} \tag{4.3i}$$

$$\frac{Z_3(t) - Z_3^I(t)}{H_3(t)} \leq \frac{Z_2(\rho_{32}(t)) - Z_2^I(\rho_{32}(t))}{H_2(\rho_{32}(t))}. \tag{4.3ii}$$

Next, consider the network structure shown in Figure 8(a). Proceeding exactly as before we replicate aggregate A_1, obtaining the subnetwork shown in Figure 8(b) with aggregates A_1 and $A_{1'}$. The system of constraints for the equivalent network is

$$F_1^{(1,2)}(Z_1) \geq Z_2 \tag{4.4i}$$

$$F_1^{(1',3)}(Z_{1'}) \geq Z_3 \tag{4.4ii}$$

$$Z_1 = Z_{1'} \tag{4.4iii}$$

Following our earlier argument, the constraints for the network system in Figure 8(b) must be

$$F_1^{(1,2)}(Z_1) \geq Z_2 \tag{4.5i}$$

$$F_1^{(1,3)}(Z_2) \geq Z_3 \tag{4.5ii}$$

From (4.5), we see that aggregate A_1 produces two distinct intermediate products, one for aggregate A_2 and the other for aggregate A_3. The derivation of (4.5) makes no assumption about the choice for the aggregate dynamic production function. Substituting our choice (3.17), the constraints become

$$\frac{Z_2(t) - Z_2^I(t)}{H_2(t)} \geq \frac{Z_1(\rho_{21}(t)) - Z_1^I(\rho_{21}(t))}{H_1(\rho_{21}(t))} \tag{4.6i}$$

$$\frac{Z_3(t) - Z_3^I(t)}{H_3(t)} \geq \frac{Z_1(\rho_{31}(t)) - Z_1^I(\rho_{31}(t))}{H_1(\rho_{31}(t))} \tag{4.6ii}$$

4.2. Aggregations of the Basic Aggregate Network Structure

We now consider the network structure shown in Figure 9(a). Here, the task is to develop a set of constraints modeling the possible choices for Z_3.

given the choices for Z_1 and Z_2. Compare Figure 9(a) to Figure 7(a): at the aggregate level the subnetworks appear to be the same; at the detailed level they are fundamentally different. In Figure 9(a) each detailed activity within aggregate 3' receives input from a detailed activity within aggregates 1 or 2 but *not* both, in contrast to the network in Figure 7(a). The outputs of aggregates 1 and 2 are used by different portions of aggregates 3'. A set of constraints other that (4.2) must be developed.

Consider Figure 9(b). Aggregates A_3 and A_4 are the *sub-aggregates* of $A_{3'}$ which use input from aggregates A_1 and A_2, respectively. From the perspective of Figure 9(b) the subnetwork in Figure 9(a) can be viewed as the result of *sequential* aggregation. First, aggregates A_1, A_2, A_3, and A_4 were formed; second, aggregates A_3 and A_4 were further aggregated into "super" aggregate $A_{3'}$. If the second iteration had not been performed, then the system of constraints for the first level of aggregation would have been

$$F_1^{(1,3)}(Z_1) \geq Z_3 \qquad (4.7\text{i})$$
$$F_2^{(2,4)}(Z_2) \geq Z_4 \qquad (4.7\text{ii})$$

By aggregating a second time we no longer know how $Z_{3'}$ is decomposed into Z_3 and Z_4. (If we knew this information, then we would just use (4.7).) We must therefore constrain the choices for $Z_{3'}$ as a function of the output curves $F_1^{(1,3)}$ and $F_2^{(2,4)}$.

Let α_3 and α_4 denote subaggregate A_3's and A_4's percentage of the total resources required by super-aggregate $A_{3'}$, respectively. It may be readily verified that each $Z_{3'} \in \hat{\Sigma}_{3'}$ is expressible as $\alpha_3 Z_3 + \alpha_4 Z_4$ for some $Z_3 \in \hat{\Sigma}_3$, $Z_4 \in \hat{\Sigma}_4$. We convexify the constraints (4.7i) and (4.7ii) in the obvious way to obtain

$$\alpha_3 F_1^{(1,3)}(Z_1) + \alpha_4 F_2^{(2,4)}(Z_2) \geq Z_3 \qquad (4.8)$$

Every choice for Z_1, Z_2, Z_3, and Z_4 feasible in (4.7) will be feasible in (4.8). For this reason we use (4.8) to model the dependence relationships for the

subnetwork shown in Figure 9a. Constraint (4.8) may be interpreted as follows: aggregates A_1 and A_2 produce the *same* product which is input by super-aggregate $A_{3'}$; the total supply of this product from A_1 and A_2 determines the progress which can be made by $A_{3'}$.

As before, the resulting inventory balance constraint is independent of the choice for the basic production function. Substituting our choice (3.17) into (4.8) and simplifying, we obtain

$$\frac{Z_{3'}(t) - Z_{3'}^I(t)}{H_{3'}(t)} \leq \frac{\alpha_3 H_3(t)}{H_{3'}(t)} \cdot \frac{[Z_1(\rho_{31}(t)) - Z_1^I(\rho_{31}(t))]}{H_1(\rho_{31}(t))} + \frac{\alpha_4 H_4(t)}{H_{3'}(t)} \cdot \frac{[Z_2(\rho_{42}(t)) - Z_2^I(\rho_{42}(t))]}{H_2(\rho_{42}(t))}. \tag{4.9}$$

Note that when $\alpha_3 = 1$, $H_3(t) = H_{3'}(t)$ and $\alpha_4 = 0$, so that (4.9) reduces to the inventory balance equation for a basic aggregate subnetwork, as expected.

Next, we consider the network structure shown in Figure 10(a), the apparent "dual" to Figure 9(a). Referring to Figure 10(b), we can view Figure 10(a) as having been obtained via sequential aggregation: first, aggregates A_1, A_2, A_3, and A_4 were formed, and second, aggregates A_1 and A_2 were aggregated into a super-aggregate $A_{1'}$. As before, if the second iteration had not been performed the inventory balance constraints would be given by (4.7). Since the second iteration was performed we have no knowledge of how $Z_{1'}$ is decomposed into Z_1 and Z_2. Once again we must combine constraints (4.7i) and (4.7ii) in some logical way.

Since the arguments of the production functions $F_1^{\{1,3\}}$, $F_2^{\{2,4\}}$ are no longer known, it is not possible to "add" these two functions (as in the previous case) without specifying the functional form for the basic aggregate production function. Substituting our choice (3.17) into (4.7), we write

$$\frac{Z_3(\rho_{31}^{-1}(t)) - Z_3^I(\rho_{31}^{-1}(t))}{H_3(\rho_{31}^{-1}(t))} \leq \frac{Z_1(t) - Z_1^I(t)}{H_1(t)} \tag{4.10i}$$

$$\frac{Z_4(\rho_{42}^{-1}(t)) - Z_4^L(\rho_{42}^{-1}(t))}{H_4(\rho_{42}^{-1}(t))} \le \frac{Z_2(t) - Z_2^L(t)}{H_2(t)} \tag{4.10ii}$$

In (4.10), we have written the balance constraints in terms of the inverses of the time transformation maps in order to present the same time arguments in the functions for A_1 as in the functions for A_2. Proceeding as before, we let α_1 and α_2 denote the fraction of total resources consumed by $A_{1'}$ which are applied to sub-aggregates A_1 and A_2, respectively. Noting that $\alpha_1 Z_1 + \alpha_2 Z_2 = Z_{1'}$, we combine (4.10i) and (4.10ii) to obtain

$$\frac{\alpha_1 H_1(t)}{H_{1'}(t)} \cdot \frac{Z_3(\rho_{31}^{-1}(t)) - Z_3^L(\rho_{31}^{-1}(t))}{H_3(\rho_{31}^{-1}(t))} + \frac{\alpha_2 H_2(t)}{H_{1'}(t)} \cdot \frac{Z_4(\rho_{42}^{-1}(t)) - Z_4^L(\rho_{42}^{-1}(t))}{H_4(\rho_{42}^{-1}(t))} \le$$
$$\frac{Z_{1'}(t) - Z_{1'}^L(t)}{H_{1'}(t)}. \tag{4.11}$$

When $\alpha_1 = 1$, $H_1(t) = H_{1'}(t)$ and $\alpha_2 = 0$, so that (4.11) reduces to the inventory balance equation for a basic aggregate subnetwork, as expected.

Constraint (4.11) may be interpreted as follows: super-aggregate $A_{1'}$ produces one intermediate product which is a shared input to aggregates A_3 and A_4; the total consumption of this product by A_1 and A_2 determines the progress of $A_{3'}$ required to support their activity.

5. Comparison to the Leachman-Boysen Model

Leachman and Boysen [1985] propose aggregate activity dependence relationships expressed in terms of the "relative earliness" of the aggregates. For $t \in [E_i, L_i]$, they define the relative earliness of A_i as

$$p_i(t) = \begin{cases} 0 & t = E_i \\ \dfrac{Z_i(t) - Z_i^L(t)}{Z_i^E(t) - Z_i^L(t)} & E_i < t < L_i \\ 1 & t = L_i. \end{cases} \tag{5.1}$$

For the basic aggregate network structure in which A_i precedes A_j, they require, for each $t \in [E_j, L_j]$ that

$$p_i(\rho_{ji}(t)) \geq p_j(t) \qquad (5.2)$$

where $\rho_{ji}(t)$ is chosen as the time point in $[E_i, L_i]$ which "corresponds" to time point $t \in [E_j, L_j]$ in the following sense: A vertical slice at time $\rho_{ji}(t)$ divides the area between the window curves of A_i into proportions the same as a vertical slice at time t divides the area between the window curves of A_j.

For more complex structures which are decomposable into replications of the basic aggregate network structure, multiple constraints of the form (5.2) are proposed; for structures which are aggregations of the basic aggregate network structure, constraints are proposed of the form

$$\sum_i c_{ij} p_i(\rho_{ji}(t)) \geq p_j(t) \qquad (5.3)$$

or

$$p_i(t) \geq \sum_j \bar{a}_{ij} p_j(\rho_{ji}^{-1}(t)). \qquad (5.4)$$

where (5.3) applies in the case that subaggregates of A_j are preceded by different aggregates, and where (5.4) applies in the case that subaggregates of A_i are succeeded by different aggregates. The coefficients c_{ij} and $\bar{a}_{ij}$ above are constants expressing the relative resource requirements of the subaggregates.

The Leachman-Boysen constraint (5.2) is identical to (3.17), utilizing the same ρ map. Their constraints in the case of replicated structures also coincide with our derived constraints. However, the Leachman-Boysen constraints are not the same as our derived constraints in the case of aggregated structures. In terms of relative earliness, our constraint (4.9) may be rewritten as

$$\left[\frac{\alpha_3 H_3(t)}{H_{3'}(t)}\right] p_1(\rho_{31}(t)) + \left[\frac{\alpha_4 H_4(t)}{H_{3'}(t)}\right] p_2(\rho_{42}(t)) \geq p_{3'}(t). \qquad (5.5)$$

Since $\alpha_3 H_3(t) + \alpha_4 H_4(t) = H_{3'}(t)$, equation (5.5) reveals that in our model the relative earliness of $A_{3'}$ at time t is bounded by a *time-varying convex combination* of the relative earliness of A_1 and of A_2 *at respective time points* $\rho_{31}(t)$,

$p_{42}(t)$. Comparing our model to (5.3), we see that Leachman-Boysen uses *constant* weights, which is more approximate; moreover, they use $\rho_{3'1}(t)$, $\rho_{3'2}(t)$ instead of $\rho_{31}(t)$, $\rho_{42}(t)$. Essentially, our model would reduce to the Leachman-Boysen model only if sub-aggregates A_3 and A_4 were identical.

In terms of relative earliness, our constraint (4.11) may be rewritten as

$$\left[\frac{\alpha_1 H_1(t)}{H_{1'}(t)}\right] p_3(\rho_{31}^{-1}(t)) + \left[\frac{\alpha_2 H_2(t)}{H_{1'}(t)}\right] p_4(\rho_{42}^{-1}(t)) \leq p_{1'}(t). \tag{5.6}$$

Since $\alpha_1 H_1(t) + \alpha_2 H_2(t) = H_{1'}(t)$, equation (5.6) reveals that in our model the relative earliness of A_i bounds a *time-varying convex combination* of the relative earliness of A_3 and of A_4 at *respective time points* $\rho_{31}^{-1}(t)$, $\rho_{42}^{-1}(t)$. Comparing to (5.4), once again, we find that Leachman-Boysen uses constant weights, which is more approximate; moreover, once again they use different ρ maps. Essentially, our model reduces to the Leachman-Boysen model only if sub-aggregates A_1 and A_2 are identical.

6. A Linear Programming Approach to Multi-Project Planning

Constraints of the form (2.4), (3.17), (4.9) and (4.11) serve to restrict the possible choices for the intensities (i.e., the resource applications) of the aggregate activities of each project. Adding standard resource capacity constraints, the overall set of constraints is expressed as a set of linear constraints on discrete-time intensity variables for each aggregate activity of each project. The set of constraints models the set of feasible resource allocations to the projects. Using an appropriate objective function, a linear program is solved to compute optimal intensities and hence optimal resource allocations.

The resource capacities for each project are defined by summming up the resource allocations in the l.p. solution to the project level. To schedule the detailed activities of each project, we propose the application of a resource-constrained scheduling technique (Wiest [1967], Talbot [1982] or Dincerler

[1984]). Note that aggregate intensities never need to be disaggregated; they simply serve as model elements for describing the set of feasible resource allocations to the projects. The l.p. model determines intelligent resource allocations to each project so that conventional project scheduling techniques can be applied to each project.

7. Summary and Concluding Remarks

An aggregate model of project-oriented production has been developed for use in multi-project resource allocation. The model has been derived from reasonable axioms concerning the production function of an aggregate and from basic principles such as inventory balance and resource conservation. By defining basic assumptions and proceeding formally, we obtain improvements to a previously-proposed aggregate model.

We remark that the three-phase approach of elevating a model from a computational form to a continuous-time framework, rigorously analyzing it relative to basic assumptions and elementary principles, and then restoring it to a computational form can improve production models in many contexts. Even simple, familiar models such as standard linear programming formulations with time lags, material requirements planning, and critical path scheduling can be reformulated more accurately or more generally with this approach. See Hackman and Leachman [1986].

Acknowledgements

ment in our presentation also resulted from two thoughtful referee reports. A portion of the material in this paper was adapted from Chapter 4 of Hackman (1984).

References

Archibald, R. D. 1976. *Managing High-Technology Programs and Projects*, John Wiley and Sons, New York.

Boysen, J. 1982. "An Aggregate Project Model for Resource Allocation within Multiproject Construction Systems," Ph.D. Dissertation, College of Engineering, University of California, Berkeley.

Burman, P. J. 1972. *Precedence Networks for Project Planning and Control*, McGraw-Hill, New York.

Dalebout, D. 1983. "A Study of Dependence Relationships Between Aggregate Activities in a Project-Oriented Production System," M.S. Thesis, College of Engineering, University of California, Berkeley.

Dincerler, A. 1985. "Project Scheduling in Project-Oriented Production Systems," ORC Report 85-11, Operations Research Center, University of California, Berkeley.

Eardley, V. J. 1969. "Production of Multi-Level Critical Path Networks," Dept. of Civil Engineering, University of Illinois, Urbana.

Hackman, S. T. 1984. "A General Model of Production: Theory and Application," ISyE Report C-84-2, School of Industrial and Systems Engineering, Georgia Institute of Technology.

Hackman, S. T. and R. C. Leachman. 1986. "A General Framework for Modeling Production," ORC Report 86-9, Operations Research Center, University of California, Berkeley, submitted to *Management Science*.

Harris, R. B. 1978. *Precedence Arrow Networking Techniques for Construction*,

John Wiley and Sons, New York.

Leachman, R. C. and J. Boysen. 1982. "Aggregate Network Model of Ship Overhaul," ORC Report 82-2, Operations Research Center, University of California, Berkeley.

Leachman, R. C. and J. Boysen. 1985. "An Aggregate Model for Multi-Project Resource Allocation," in *Project Management: Methods and Studies*, B. V. Dean, ed., North Holland, Amsterdam.

Parikh, S. C. and W. S. Jewell. 1965. "Decomposition of Project Networks," *Management Science* 11, 444-459.

Talbot, F. B. 1982. "Resource-Constrained Project Scheduling with Time-Resource Tradeoffs: The Nonpreemptive Case," *Management Science*, 28, 1197-1210.

Vlach, J. 1969. "Some Results of Connecting (Aggregating) Networks in Building," in *Internet 2: Project Planning by Network Analysis*, North Holland, Amsterdam.

Wiest, J. D. 1967. "A Heuristic Model for Scheduling Large Projects with Limited Resources," *Management Science*, 13, 1120-1148.

Repair Aggregate
A_i

Re-Install Aggregate
A_j
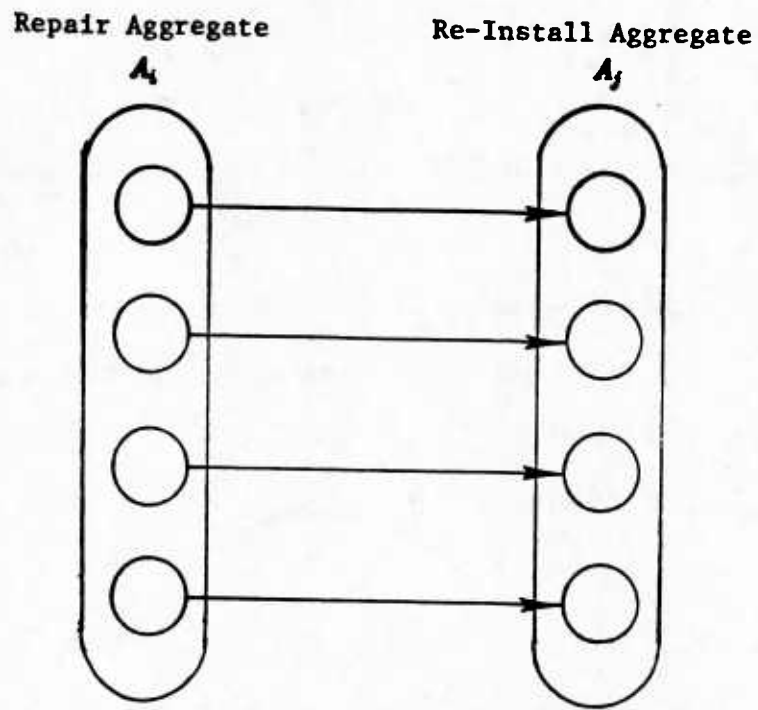
Figure 1

A Simple Example of a Basic Aggregate Network Structure

DETAILED ACTIVITY	REPAIR AGGREGATE A_i				RE-INSTALL AGGREGATE A_j			
l	ES_l	LS_l	d_l	α_{li}	ES_l	LS_l	d_l	α_{lj}
1	0	6	3	1/4	3	9	4	1/3
2	1	4	4	1/3	5	8	3	1/4
3	2	6	4	1/3	6	10	3	1/12
4	3	5	3	1/12	6	8	4	1/3

TABLE 1

NUMERICAL DATA FOR THE SIMPLE EXAMPLE
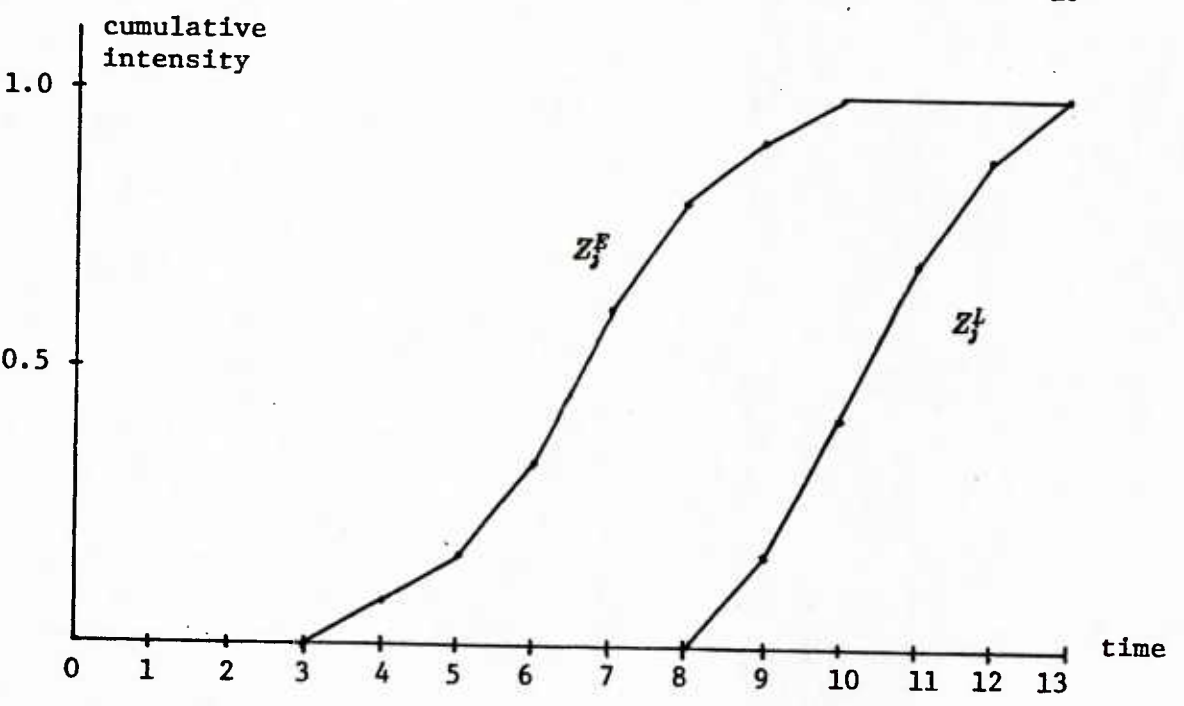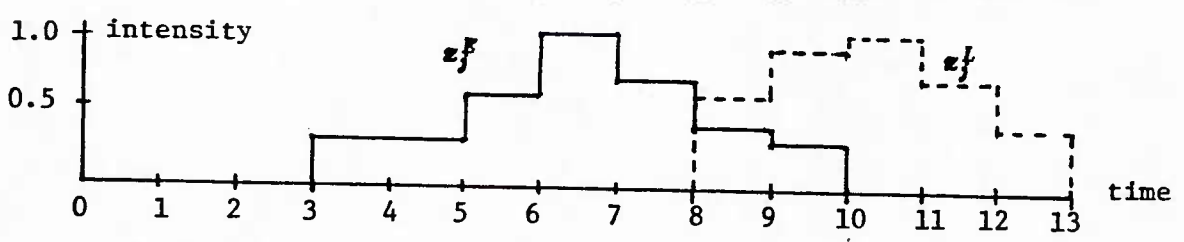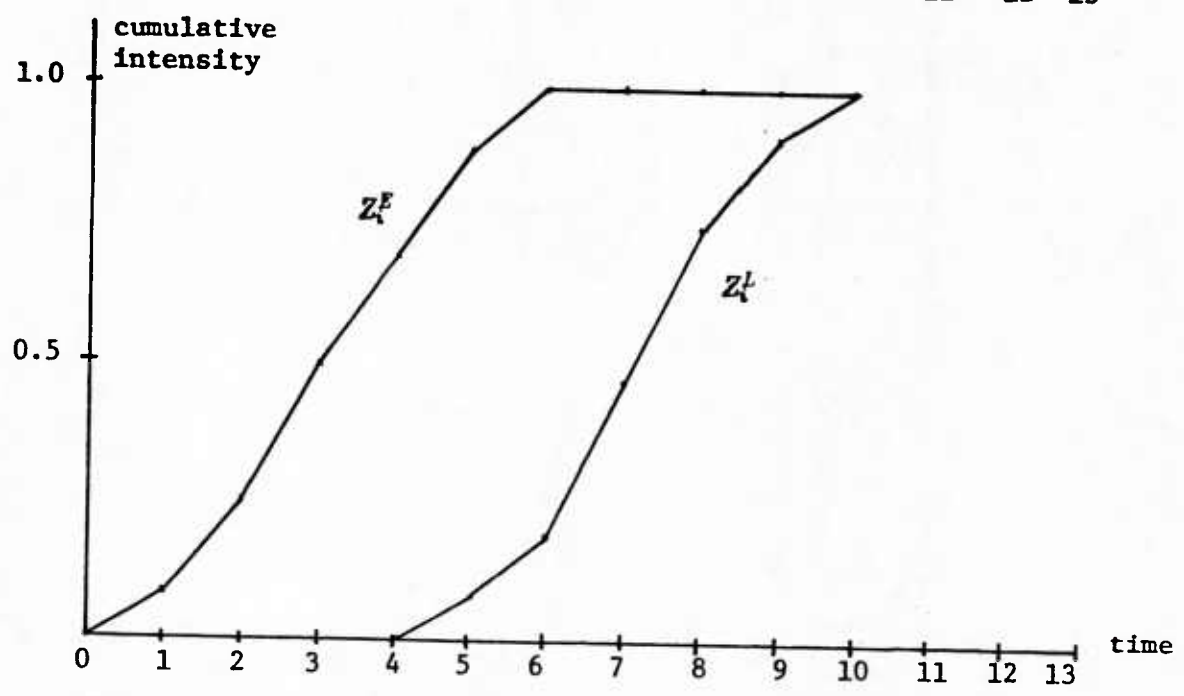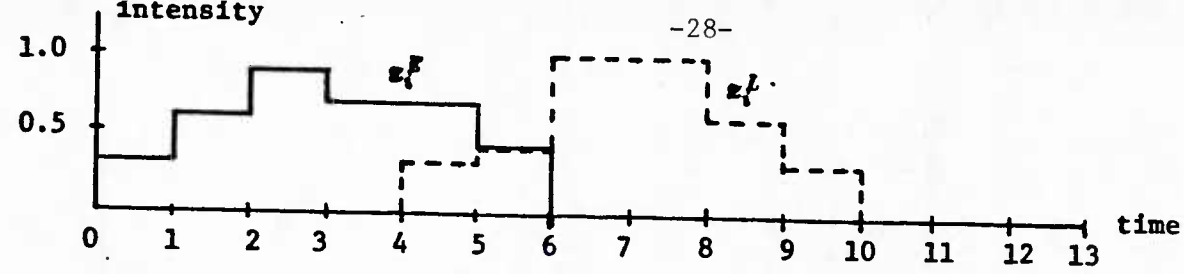
Figure 2

Boundary Curves for the Simple Example
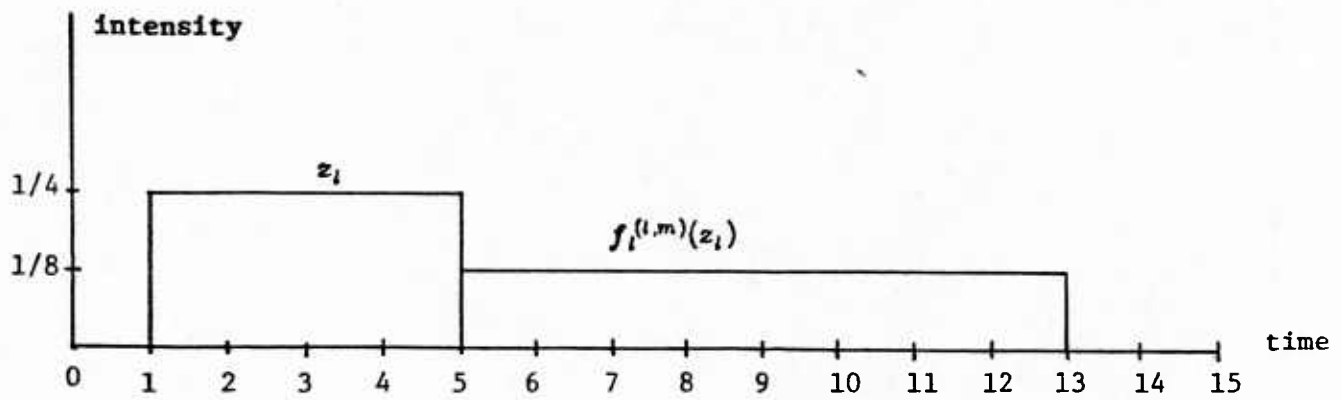
(a)
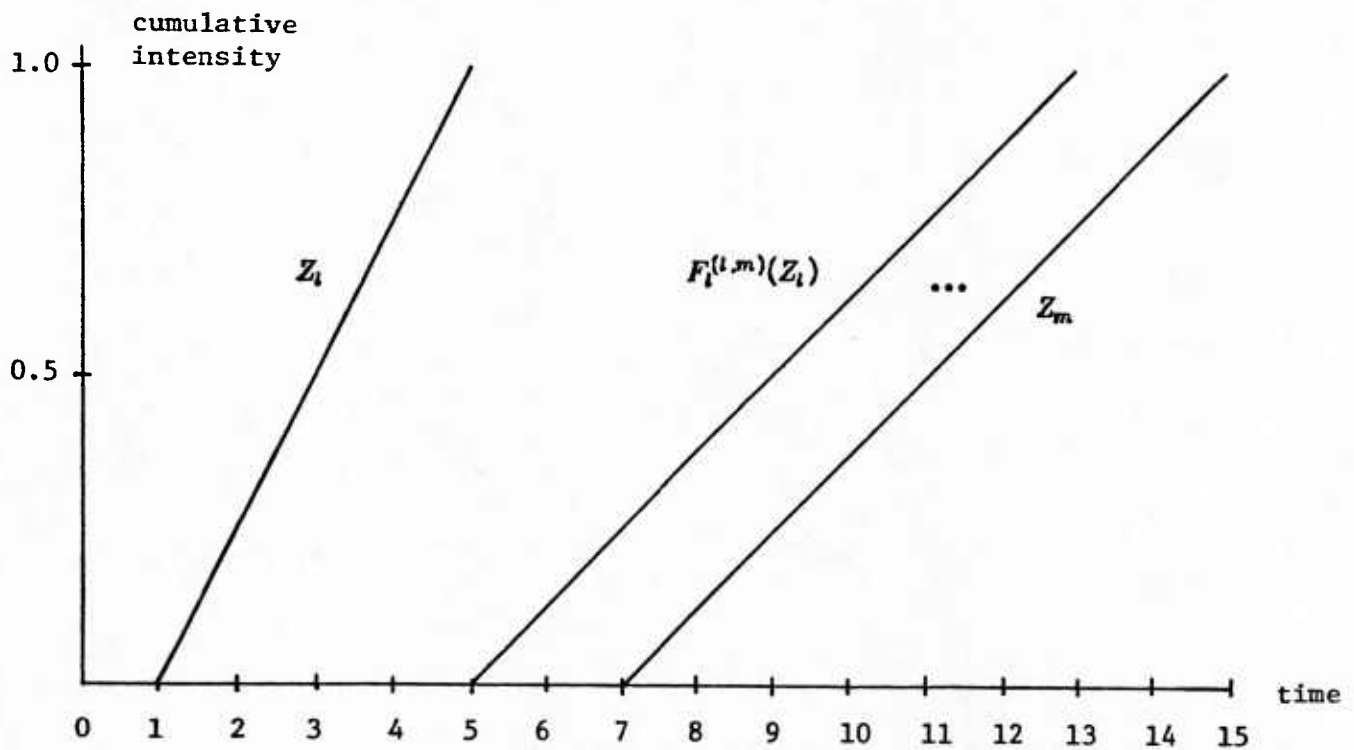
(b)

Figure 3

Production Function for CPM
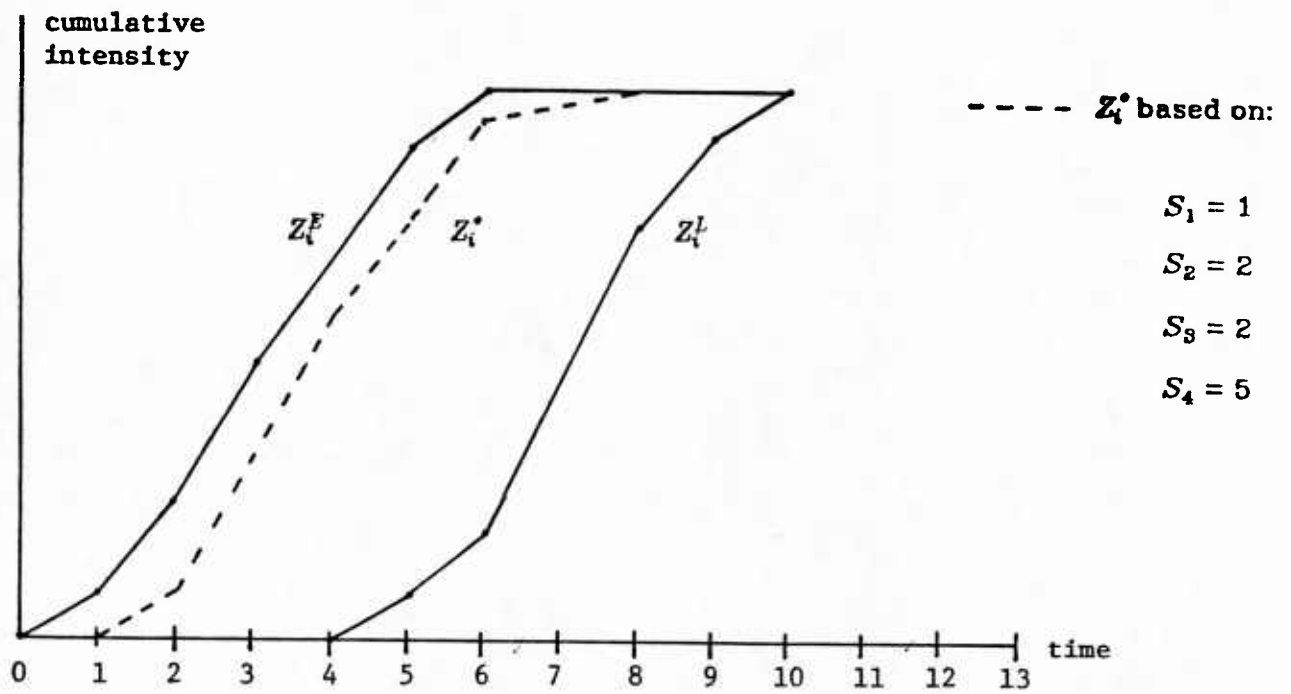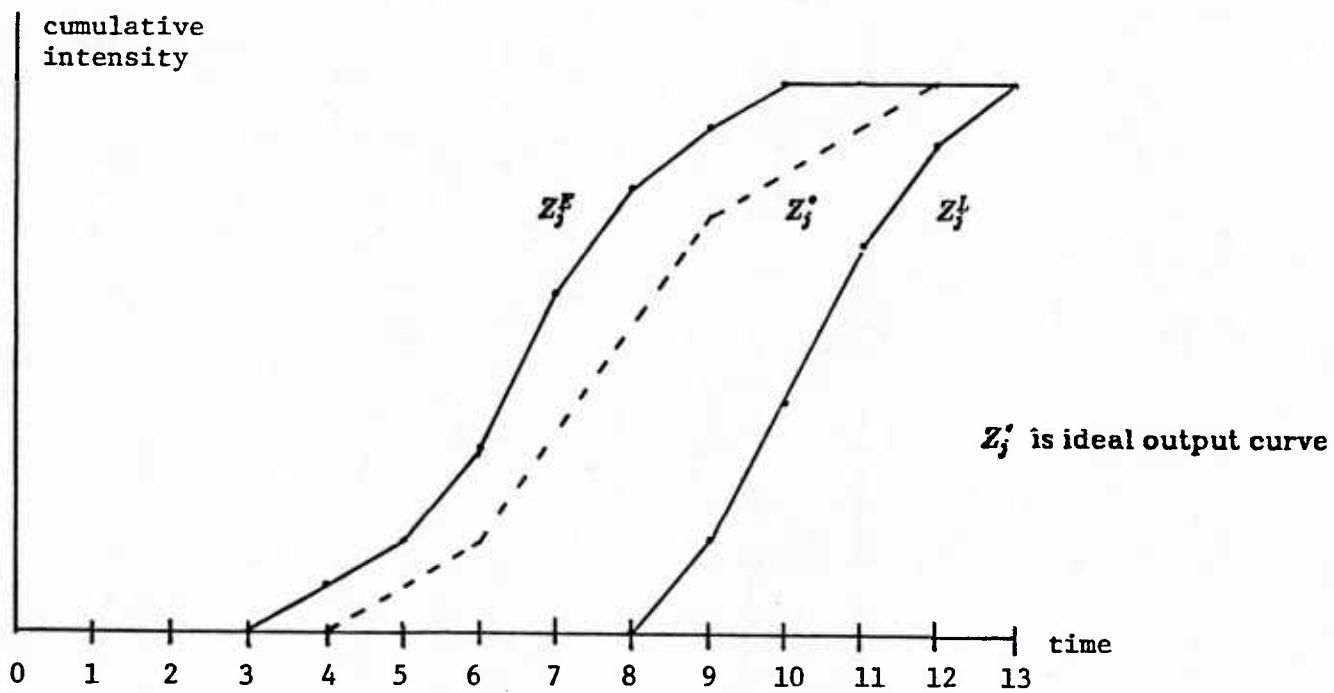
Figure 4

Ideal Aggregate Production Function
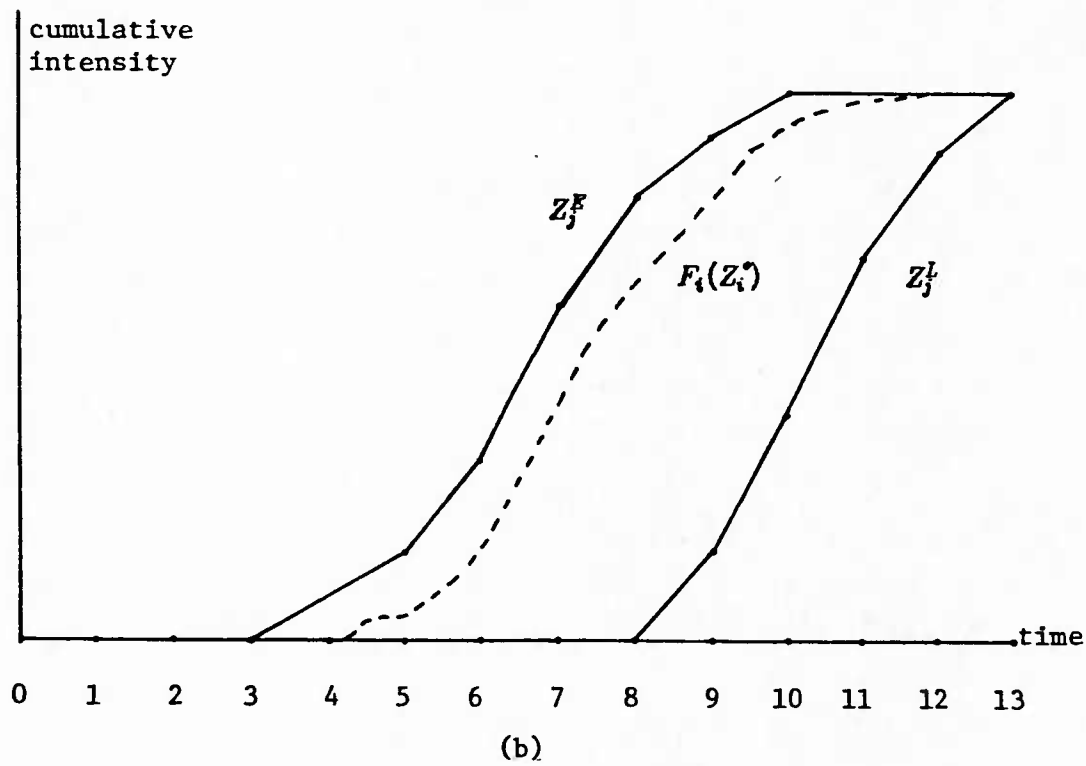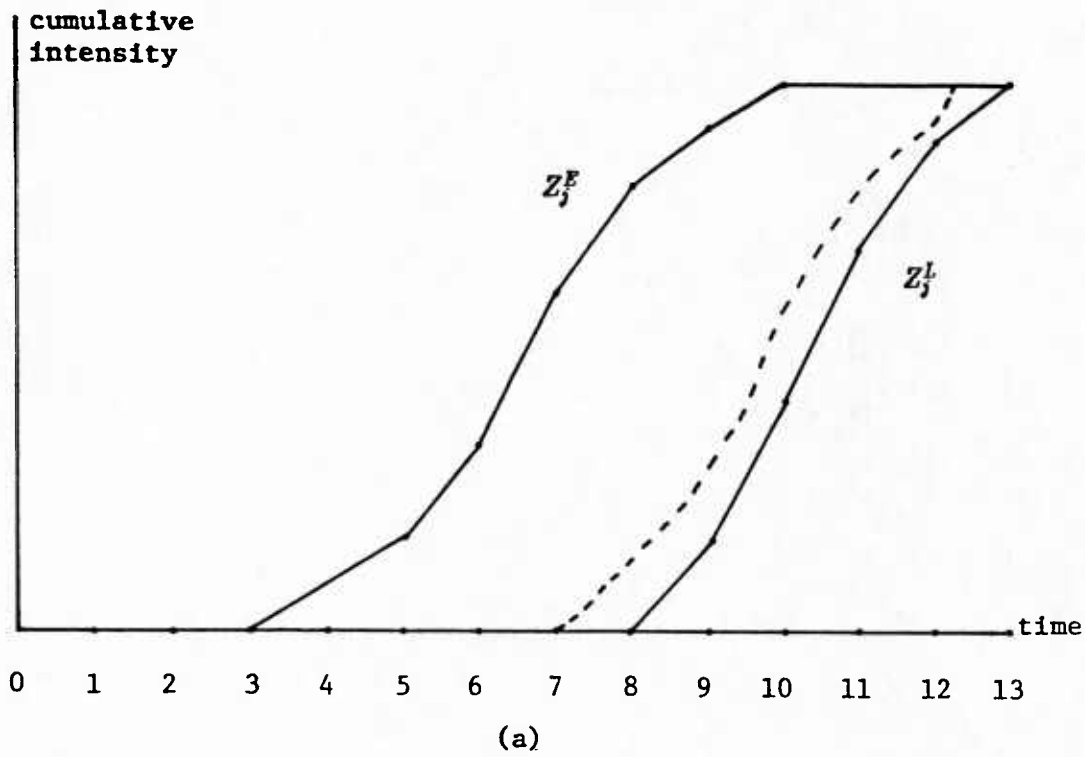
Figure 5

Proposed Output Curves for the Simple Example
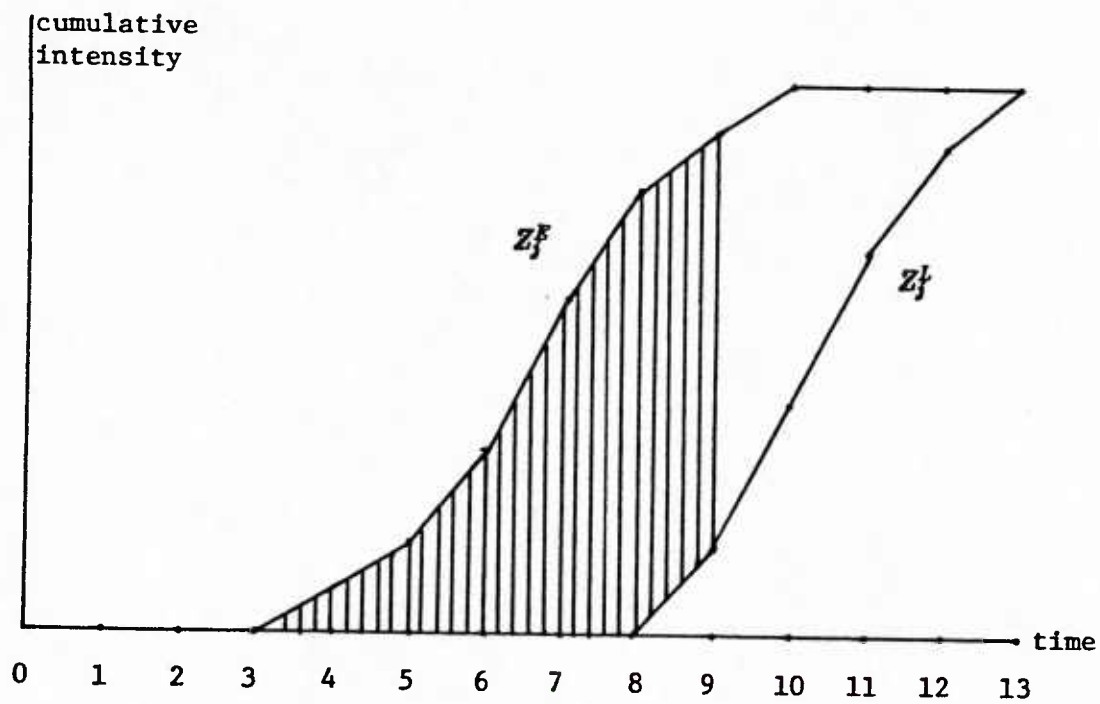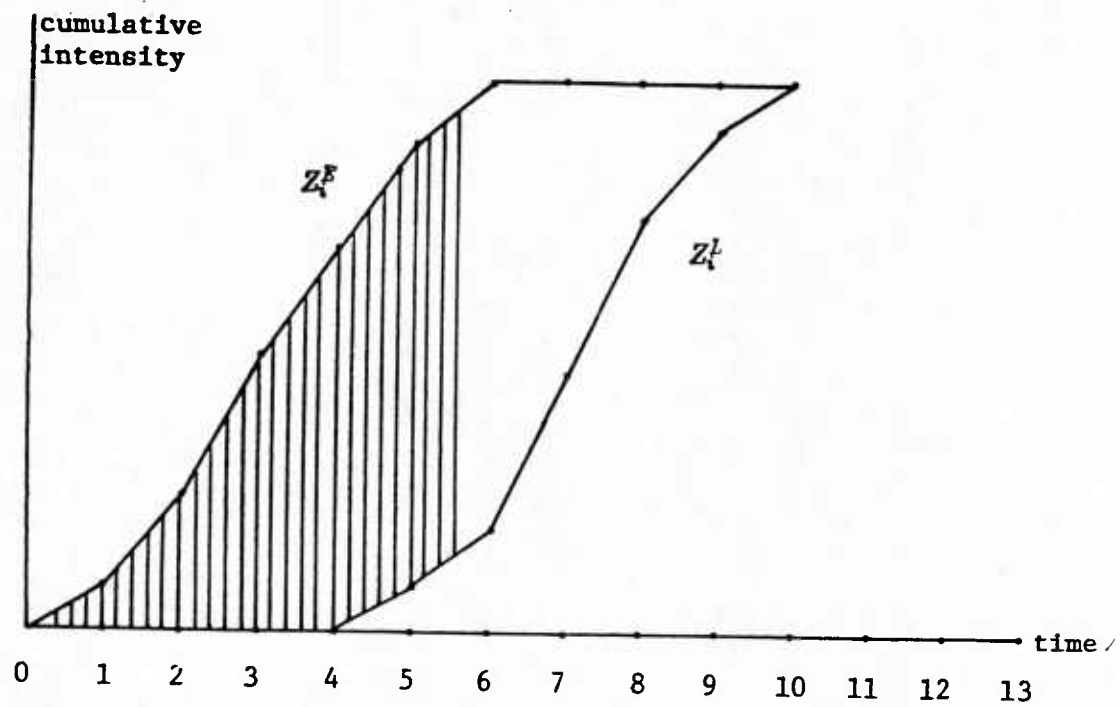
$$\rho_{ji}(\theta) = 5.6$$

Figure 6

Geometric Property of the ρ Map
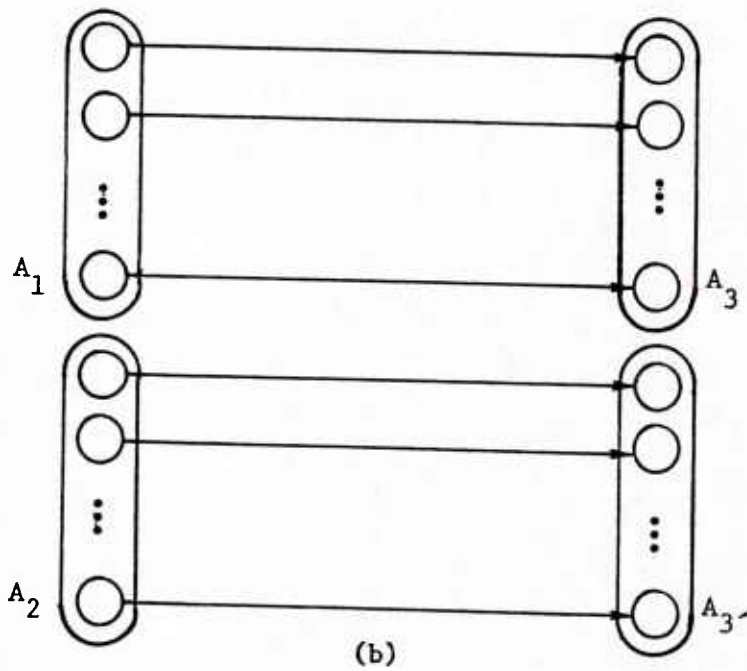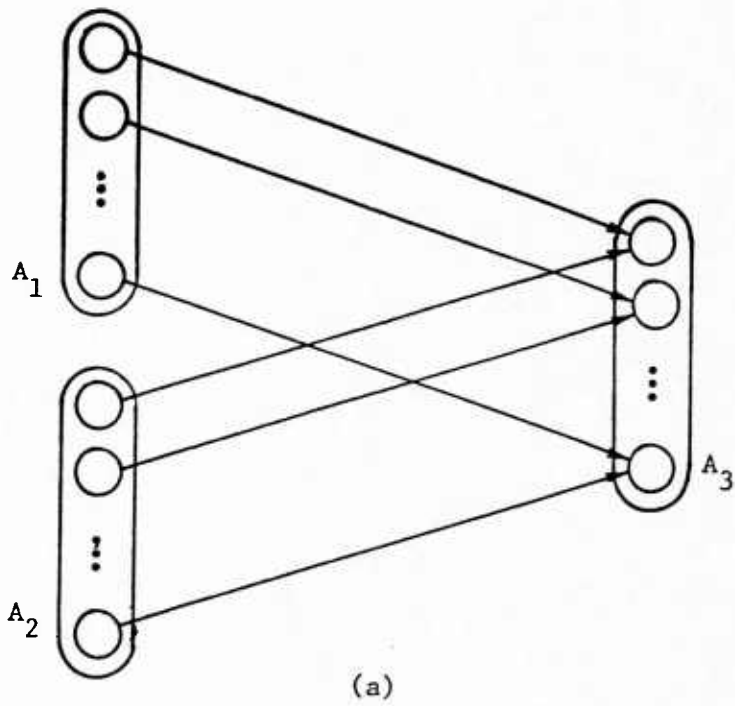
Figure 7

Replicated Aggregate Network Structure:
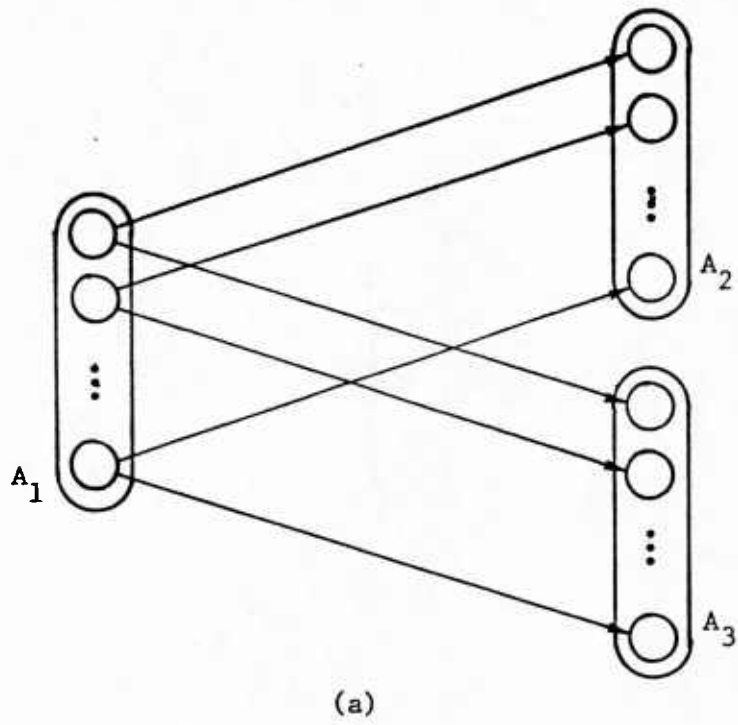The Case of Multiple Predecessors

(a)

(b)
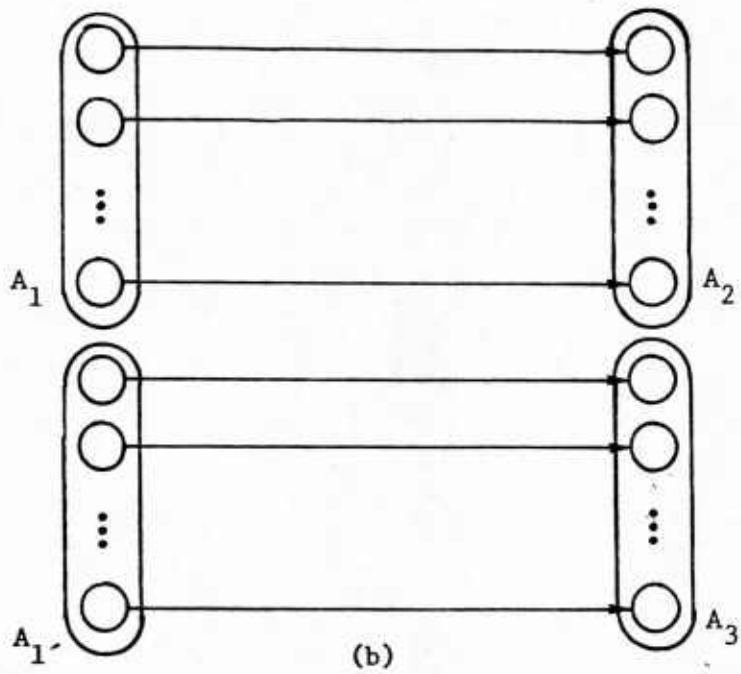
Figure 8

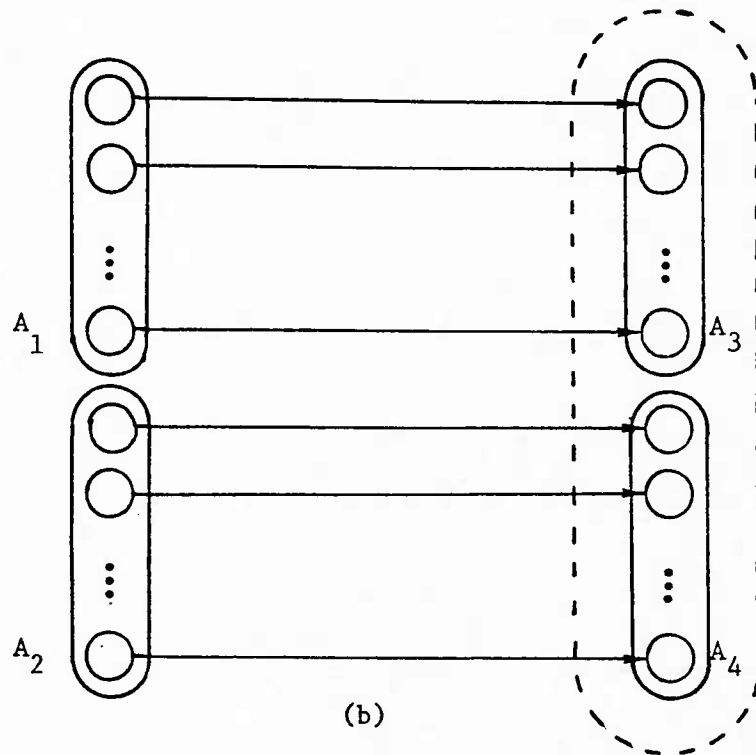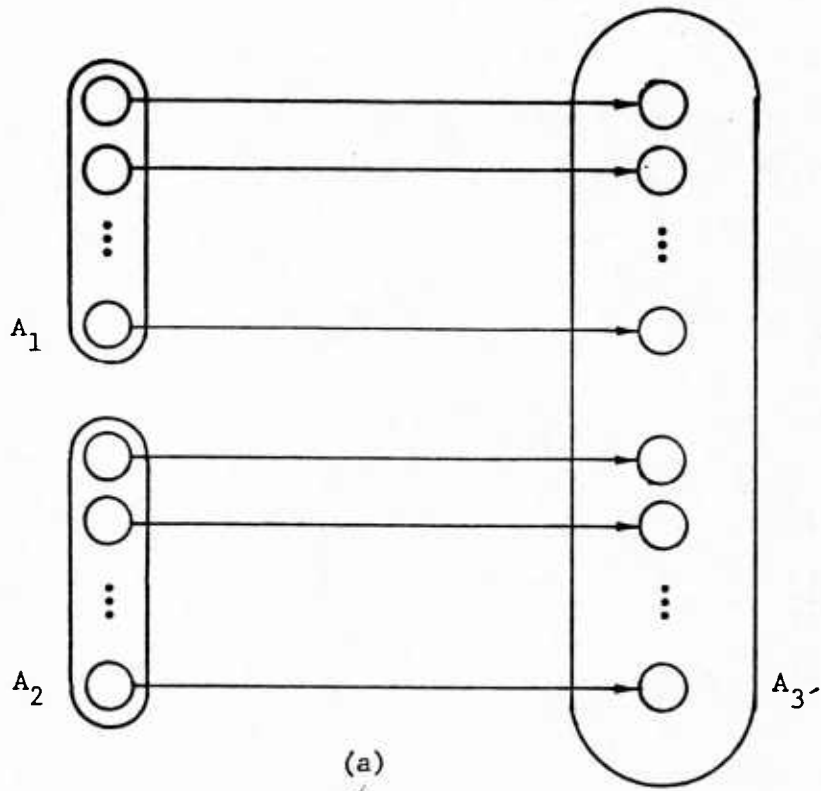Replicated Aggregate Network Structure:
The Case of Multiple Successors

Figure 9

Aggregation of Basic Structures:
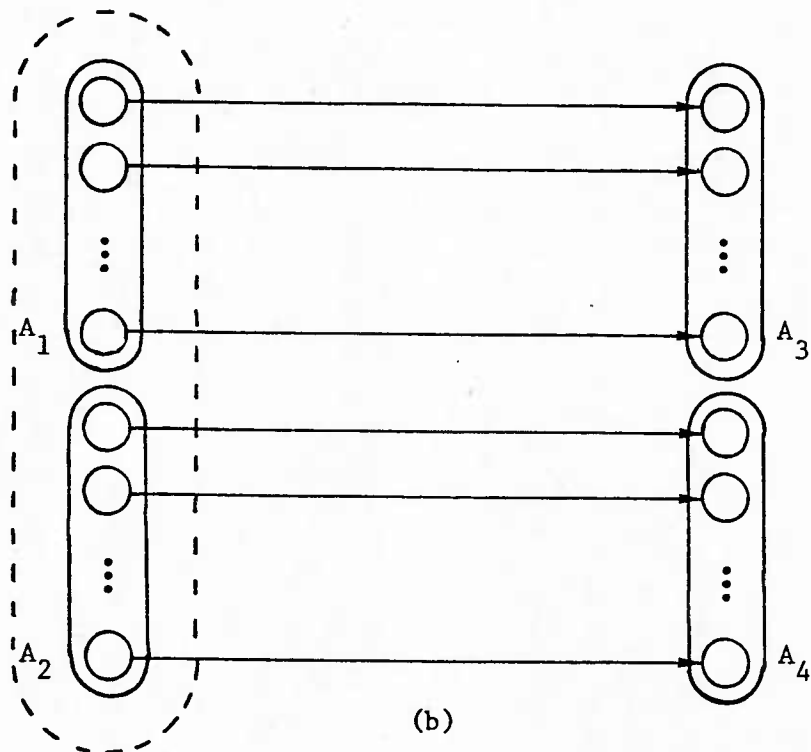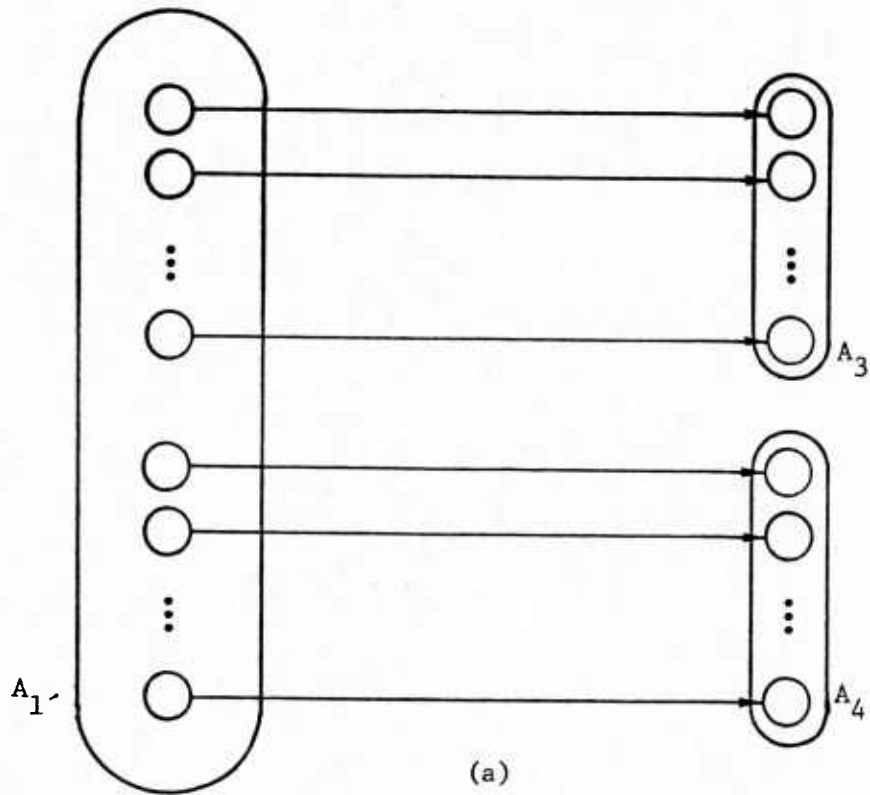The Case of Multiple Predecessors

(a)

(b)

Figure 10

Aggregation of Basic Structures:
The Case of Multiple Successors